Camera pose estimation in unknown environments using a sequence of wide-baseline monocular images

HOSEINI, Seyyed Ali and KABIRI, Peyman <http://orcid.org/0000-0001-5143-0498>

*Journal of **AI** and **D**ata **M**ining*
Vol 6, No 1, 2018, 93-103

Camera Pose Estimation in Unknown Environments using a Sequence of Wide-Baseline Monocular Images

S. A Hoseini and P. Kabiri*

Department of Computer Engineering, Iran University of Science and Technology, Tehran, Iran.

Received 02 November 2016; Revised 31 January 2017; Accepted 10 May 2017

**Corresponding author: peyman.kabiri@iust.ac.ir (P. Kabiri).*

Abstract

In this work, a feature-based technique is proposed for the camera pose estimation in a sequence of wide-baseline images. Camera pose estimation is an important issue in many computer vision and robotics applications such as augmented reality and visual SLAM. The developed method can track captured images taken by a hand-held camera in room-sized workspaces with a maximum scene depth of 3-4 m. This system can be used in unknown environments with no additional information available from the outside world except in the first two images used for initialization. Pose estimation is performed using only natural feature points extracted and matched in successive images. In wide-baseline images, unlike consecutive frames of a video stream, displacement of the feature points in consecutive images is notable, and hence, cannot be traced easily using the patch-based methods. To handle this problem, a hybrid strategy is employed to obtain accurate feature correspondences. In this strategy, first, initial feature correspondences are found using the similarity between their descriptors, and then the outlier matchings are removed by applying the RANSAC algorithm. Further, in order to provide a set of required feature matchings, a mechanism based on the sidelong result of robust estimator is employed. The proposed method is applied on indoor real data with images in VGA quality (640 × 480 pixels), and on average, the translation error of camera pose is less than 2 cm, which indicates the effectiveness and accuracy of the developed approach.

Keywords: *Camera Pose Estimation, Feature Extraction, Feature Correspondence, Bundle Adjustment, Depth Estimation.*

1. Introduction

Camera pose estimation is one of the key issues in computer vision. In many applications, it is critical to know where the camera is located. The accurate and robust estimation of the camera position and orientation is essential for a variety of applications including 3D reconstruction, augmented reality, and visual Simultaneous Localization and Mapping (visual SLAM).

Camera tracking for a sequence of video frames is exactly the problem of camera pose estimation for each frame. For the adjacent frames of a video sequence, the camera pose has a negligible change. Moreover, the motion vector of the scene features between successive frames can be discovered using a simple patch-based similarity measure. Conversely, for wide-baseline sequences, estimation of the motion vector for feature points is not a simple task. In the computer vision literature, wide-baseline images refer to a condition where the distance between the camera center for adjacent images is noticeable or the camera orientation changes remarkably. Moreover, once the internal parameters of the camera change (i.e. zooming), the resulting images simulate the wide-baseline situation. In contrast, when the camera motion is smooth, the camera center for adjacent frames are close to each other. This leads to a negligible displacement of the points of interest in consecutive frames. This case is usually referred to as narrow-baseline. There are situations where it is more reasonable to estimate camera pose for a sequence of wide-baseline images. Reducing the computational cost, some video tracking algorithms are based upon the selected key-frames. These key-frames often form a sequence of wide-baseline frames. Also for low-quality images (like VGA), a quick movement of camera may result in a sequence of several blurred frames. Feature tracking along blurred frames is a challenging task. Hence, it is better to ignore them. The wide-baseline situation is resulted due to ignoring the successive frames. Furthermore, using a limited number of images may considerably speed up the reconstruction process.

Nevertheless, it is worth noting that the wide-baseline setting often allows a more accurate depth calculation. An increase in the depth accuracy is due to a larger, and hence, more reliable measurable disparities in the images. However, there are configurations (i.e. when the camera has rotation about its optical axis) in which the motion vector for tracked features varies significantly. In these situations, some features may introduce small disparities, while others have remarkable displacements.

For a wide-baseline case, determining the feature correspondences is a challenging task. However, with the advent of local descriptors, finding similar regions within the images taken from different viewpoints became promising. In the subsequent sections, some outstanding descriptor-based feature extractors will be introduced.

Occlusion is yet another problem for the wide-baseline case. Some features may be occluded when the camera undergoes remarkable changes in viewpoint. Occlusion usually reduces the number of matched features. It may also lead to false matchings. Generally, mitigating the undesired effects of occlusion or any problem that produces false matchings, robust estimators such as Least Median (LMed) [1] or Random Sample Consensus (RANSAC) [2] is employed. As a result, the incorrect feature correspondences are eliminated.

In this paper, the problem of camera pose estimation for a sequence of wide-baseline monocular images is addressed. The images are captured with a single camera from adjacent locations in such a way that the overlapping regions in consecutive images are adequate for obtaining the common features. On the other hand, the area of overlapping regions is not large enough to provide the feature point correspondences through correlation windows.

Camera pose estimation and 3D reconstruction are tightly coupled, i.e. to estimate the parameters of the camera motion, it is necessary to have sufficient information about the 3D structure of the scene. On the other hand, triangulating depth of newly extracted features, it is necessary to have the camera pose from two or more views available.

1.1. Pose parameters

As depicted in figure 1, a moving camera captures images of the environment from arbitrary positions. For each view, pose of the camera is composed of two parts: the rotation matrix $R \in \mathbf{R}^{3\times3}$, which is an orthogonal matrix with $\det(R) = 1$ that describes the orientation of camera, and the translation vector $t \in \mathbf{R}^3$ that indicates the distance between the origin of camera coordinate system and the world coordinate system. Accordingly, (1) is established for every 3D point in the scene [3].

$$X_c = RX_w + t \qquad (1)$$

X_c, X_w are the coordinates of the 3D point with respect to the camera and world coordinate systems, respectively.

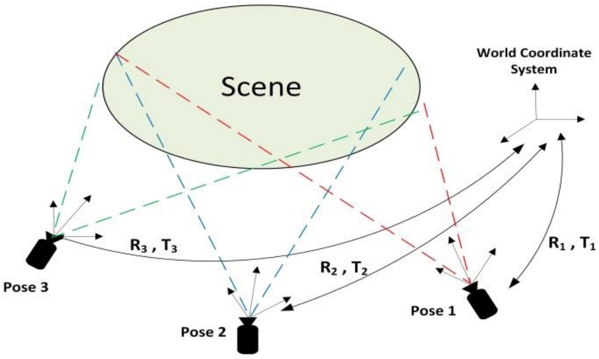

Figure 1. Multi-view camera pose estimation.

The structure of this paper is as what follows. The related works are discussed in Section 2. In Section 3, the proposed approach will be explained in details. The experimental results are presented in Section 4. Finally, the conclusions and future works are included in Section 5.

2. Relative work

Camera tracking or estimation of camera pose parameters for sequence of video frames that represents the narrow-baseline situation has been widely studied. In this research area, two main solution categories exist, i.e. Structure from Motion (SfM) and filtering. The SfM approach uses the epipolar geometry principles to solve the problem. Often to refine the estimated parameters of the camera and the depth of feature points, an additional optimization stage is required. Bundle Adjustment (BA) [4] and pose map [5] are two main strategies used for this purpose. Parallel Tracking and Mapping is a prominent work that uses BA to optimize the estimated camera Pose [6]. Some researchers have employed the pose map optimization technique to improve the accuracy of the estimated camera trajectory [7, 8].

In filtering approaches, the problem is cast in the shape of a dynamic system in which the camera pose parameters constitute the internal state of the system. Furthermore, the state transition of the system is usually a non-linear relation based on the physical nature of rigid body motion in 3D space. Meanwhile, the projection of 3D features on image plane using current rotation and translation of camera introduces the observation model of the system. Mostly, due to the non-linear nature of transition and observation model, variants of Kalman filter such as Extended Kalman Filter (EKF) and Unscented Kalman Filter (UKF) are used for pose estimation [9, 10]. Particle Filter (PF) is another solution in the context of dynamic systems, which is utilized for this purpose [11-13]. As opposed to the narrow-baseline case, the filtering techniques for wide-baseline are not easily applicable. This is due to the fact that in filtering approaches, the motion model definition is usually meaningful for small changes in the system state. However, it is not the case for the wide-baseline condition. Hence, it is more realistic to exploit SFM to handle Camera pose estimation for the wide-baseline images.

In any case, the necessary information to obtain orientation and translation of the camera is a set of point correspondences in two or more views. If these correspondences are given in 3D-3D matchings, then it is the subject of absolute orientation problem that can be solved easily using closed-form solutions proposed for this problem [14-16]. When the supplied correspondences are in the form of 3D-2D matchings, then the problem is known as Perspective n Point (PnP) in computer vision literature for which Several solutions are proposed [17, 18]. Sometimes the available information is only some 2D-2D correspondences. In such circumstances, using the notion of fundamental matrix and epipolar geometry, the camera pose parameters are estimated with ambiguity. On the other hand, multiple solutions are obtained. In order to achieve a unique solution, it is necessary to have extra information about the observed scene.

It is well-known that receiving no information about the depth of extracted scene features produces drift in camera trajectory, and increases the cumulative error, i.e. for a freely-moving camera, the captured images provide information about the geometry of the scene that can be recovered up to a scale factor using the multi-view geometry. Dealing with this problem, some researches put markers or fiducials with known structures in the scene to control the cumulative error [19, 20]. Using multiple markers in the scene could also increase the accuracy of camera pose parameters [21].

Exploiting reference calibrated images is another technique for camera tracking in unknown environments [22, 23]. The calibrated images are those with known 3D coordinate for a sparse set of features. With reference images, the process of pose estimation reduces to data association between each new image and the reference images.

The two main contributions of this work are summarized as follow:

1) **Feature correspondences**. In order to provide a sufficient number of matched features, a combination of feature matchings based on similarity of feature descriptors and homography matrix is adopted.
2) **Propagation of depth information**. In order to enable the proposed system for estimation of the camera pose of each incoming image, a novel strategy is adopted to propagate depth information of already extracted features to subsequent images.

3. Proposed method

An overview of the proposed framework is initially presented in figure 2. In the proposed method, after the arrival of each new image, the process of camera pose estimation is performed in two stages, obtaining the matched features and estimation of camera pose parameters. To provide robust matchings, the extraction of salient and repetitive feature points is necessary. The feature extraction step will be elaborated in section 3.2. Thereafter, the extracted feature points should be matched with those of the previous image. The matchings obtained that are robust enough are used for estimation of the camera pose parameters. In Section 3.3, the issue of finding the feature point correspondences and refining them will be discussed. In the next step, camera pose for the current image is retrieved by utilizing the obtained correspondences. Since retrieving the camera pose parameters is based upon 3D-2D matchings, it is required that the depth of sufficient number of feature points among the obtained correspondences already estimated.

In the reported method, a collection of feature points with a known 3D coordinate is updated for each new image. We called this collection as fully active features. This means that with every new image, the newly extracted feature points that were matched in two recent images will be added to the previously collected feature set. Furthermore, estimating the pose parameters of the camera based on 3D-2D matchings, the feature points with known 3D coordinates are selected from this collection. It should be noted that the 3D position of fully active features is measured with respect to the world coordinate system.

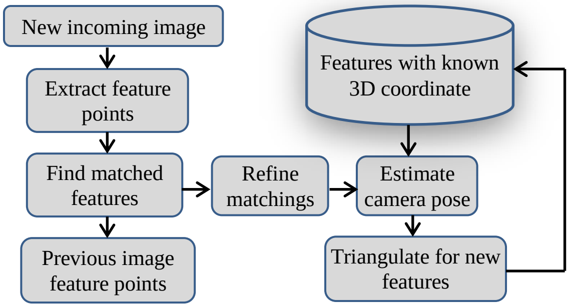

Figure 2. Overview of proposed approach.

Since the unknown parameters for camera pose estimation and depth of feature points are estimated incrementally, the associated error is accumulative. Minimizing the accumulative error, in the final step, a windowed bundle adjustment is applied to optimize the estimated pose parameters for all the input images.

In the proposed framework, there is no way to recover the depth of newly added features except using the structure of features with determined 3D position. From a set of 2D-2D matchings in two or more images, it is only possible to estimate the depth of corresponding features with a scale factor [24]. This limitation enforces the algorithm to start from a calibrated image, i.e. initially, a small amount of prior information about the scene in the form of known targets should be available. In the proposed system, a chessboard with known size is placed in front of the camera. This provides a set of feature points (corners of the chessboard cells) with known positions in the world coordinate system that allows us to estimate camera pose parameters for the first and second images. At the same time, natural features extracted and matched are triangulated using camera poses in the first and second images. Then the depth information of these features will be propagated to the subsequent images.

3.1. Wide-baseline situation

As explained earlier, in wide-baseline images, displacement of the corresponding feature points are noticeable with respect to the image size. This issue is illustrated in figure 3. The feature point displacement in two images depends upon the amount of changes in the pose parameters of the camera and the depth of the observed scene. If the camera undergoes a significant change in position or orientation for two consecutive poses, then the associated images will be less overlapped. Hence, using the traditional patch-based similarity measures such as the sum of squared differences or normalized cross-correlation are not practical for data association. This is due to the fact that these measures are convenient for small changes in the camera view, which is not the case in the wide-baseline situation. Moreover, in cases where the distance of the camera from the scene is notable, applying a slight motion to the camera results in a noticeable displacement of the feature points. The aforementioned issues in the wide-baseline condition make the problem of feature matching a challenging task.

In addition, each feature is only visible in a small number of images. This problem causes that the necessity for triangulation of newly extracted features occurs more frequently.

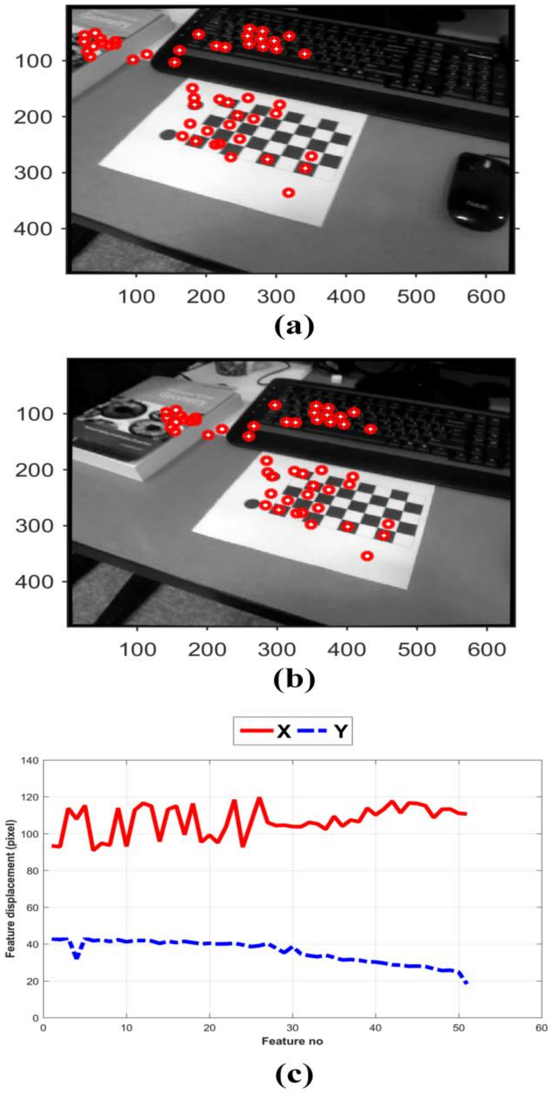

Figure 3. Wide-baseline condition (a) and (b) sparse set of feature correspondences (c) displacement of corresponding features in X and Y directions.

3.2. Feature extraction

In the proposed approach, in order to determine the relationship between images, a feature-based method is utilized. In the feature-based methods, different entities such as points, lines, region or objects can be selected as the feature. However, among them, the point features are better than the others since they are easier to detect and match. In addition, the number of detected feature points is usually more than the other types of features, and hence, it is more likely to observe them in the successive images. Many algorithms are presented to extract the feature points in images. The corners are well-known feature points. They are usually considered as the intersection of two edges. The corners may also be defined as a point where two dominant and different edge directions exist in its local neighbourhood.

Harris [25] and SUZAN [26] are famous corner detectors used in many image processing and computer vision areas such as image registration, image mosaicing, panorama stitching, and object recognition. The corners are suitable features to track in video frames since they are easily detected in successive frames and can be matched using patch-based approaches with simple similarity measures such as the sum of absolute differences or normalized cross-correlation. In contrast, for wide-baseline images, as explained earlier, image pixels undergo a remarkable displacement. Hence, it is necessary to employ features that contain descriptor. Recently, several descriptor-based feature extraction approaches have been proposed. Scale Invariant Feature Transform (SIFT) [27], Speeded-Up Robust Features (SURF) [28], and Binary Robust Invariant Scalable Keypoints (BRISK) [29] are three strong and reliable ones. They first detect the location of the feature points and then construct the associated descriptor vector from the information of image in the neighborhood of the detected location. The related descriptor vectors are invariant to scale, rotation, viewpoint, and illumination changes. This allows us to find the corresponding features using the associated descriptors by means of a simple similarity measure.

In the proposed method, the SIFT feature points were employed due to their high distinctiveness and repeatability. The generated descriptors for SIFT features are very powerful for match finding along enough overlapped images.

3.3. Feature matching and refinement

Providing accurate feature correspondences is a significant step for estimating a robust and precise camera pose parameters. As explained earlier, tracking feature points is highly susceptible to the production of incorrect matched features. Handling this problem, we require following the "detect and match" strategy to obtain the feature correspondences. In other words, initially, each incoming image SIFT features are detected, and then the presence of shared features in both the current and earlier images are matched. This task is achieved using a similarity measure between the feature descriptors. In the reported work, the cosine distance was used for this purpose, as given in (2).

$$d(D_i, D_j) = 1 - \frac{D_i^T . D_j}{\|D_i\| . \|D_j\|} \tag{2}$$

where, ||.|| denotes the L2-norm. The L2-norm of descriptor difference is also possible but it is computationally more expensive. Since the SIFT descriptors have unit norm, the similarity measure between them is calculated by a simple dot product.

The feature correspondences obtained by comparing the feature descriptors may include mismatched feature pairs, i.e. several features in the first image might be matched with a shared feature in the second image as the closest one with a minimum cosine distance. Deciding which matched feature in the second image is the correct one, the mutual consistency check is established. In order to do so, the features in the second image are paired with the features in the first one, and those that are matched in both directions are selected. This routine guarantees the mutual consistency between the matched features.

Thus the matched features may contain wrong matchings due to noise or repetitive textures. Wrong correspondences are called outliers that violate spatial consistency of image. For an accurate estimation of camera pose, these outliers should be rejected. The outlier removal is based upon the geometric constraints introduced by the motion model. RANSAC is a standard technique used for estimating the parameters of a model in the presence of outliers. The RANSAC algorithm produces the inlier correspondences as well as the parameters of the assumed model. These parameters are encoded into a 3 × 3 homography matrix (*H*), and for every feature correspondence $u_1 \leftrightarrow u_2$, the following equation holds:

$$\lambda u_2 = H u_1 \tag{3}$$

u_1, u_2 are in homogeneous coordinates, and λ is the projective scaling factor. Since *H* is computed using the inlier correspondences, given u_1 and *H*, the approximate location of u_2 in the second image can be obtained. This issue will be exploited in the next section to find the paired features in specific situations.

In figure 4, feature matchings by comparison of the descriptor vectors are marked with empty red circles. The refined matchings are also illustrated with blue asterisks surrounded by a red circle. Some matchings depicted with empty red circles are not selected after refinements, even though they are visually appeared correct matchings. It is due to the fact that during the matching refinement operation, some visually correct matchings are rejected to ensure that the selected matchings are reliable.

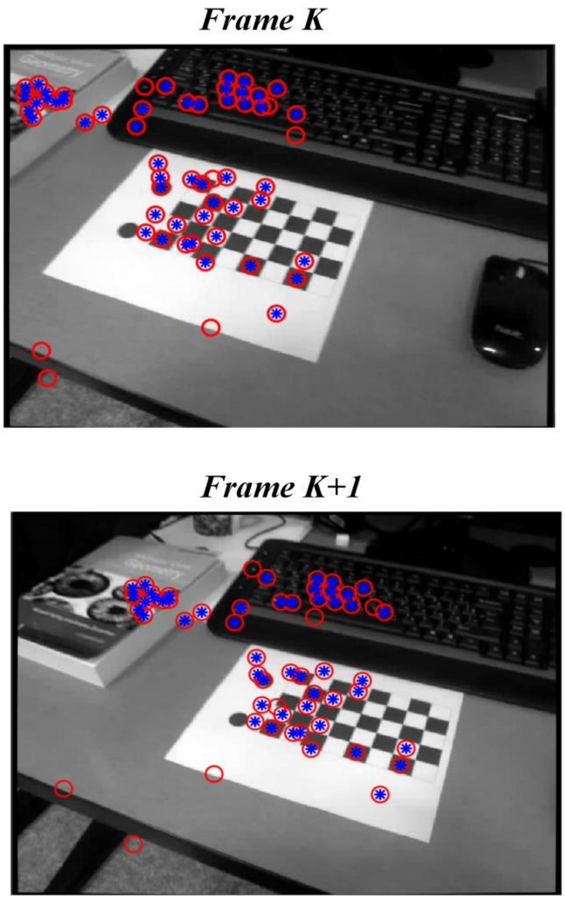

Figure 4. Feature points marked with empty red circles are output of feature matching routine, and those marked with blue asterisks surrounded in red circles are refined matchings based on RANSAC algorithm.

3.4. Providing 3D-2D matchings

In the core of our system, the pose parameters are estimated using a set of 3D-2D feature correspondences. In the previous section, it was explained how the set of paired features were adopted. Now it is necessary to provide a collection of 3D-2D feature matchings. However, in order to be able to estimate camera pose for the current image, it is required to have at least four non-coplanar 3D-2D feature matchings. Moreover, to achieve more accurate and reliable results, it is better to include more matchings.

Figure 5 shows the overall scheme of the adopted strategy to manage the obtained feature matchings to estimate the camera pose and to triangulate the partially active features. Let θ_k be the set of SIFT features extracted in the current image (I_k) and $\lambda_{k-1}, \gamma_{k-1}$ be the set of fully active and partially active features in the previous image (I_{k-1}). With fully active features, we mean those features whose depths are already estimated, and the partially active features are those with unknown depth but potential for matching with extracted features of the next image. From the matchings obtained in the current image, we define FA_k, PA_k as the set of ordered pairs of matchings established with fully (red arrows) and partially (green arrows) active features of I_{k-1}, respectively.

$$\begin{aligned} FA_k &= \{(u_1,u_2)\,|\,u_1 \in \lambda_{k-1}, u_2 \in \theta_k\}, \\ PA_k &= \{(u_1,u_2)\,|\,u_1 \in \gamma_{k-1}, u_2 \in \theta_k\} \end{aligned} \tag{4}$$

If the number of matchings in FA_k is greater than a pre-defined threshold, then the camera pose is computed using a method that will be explained in the next section. Immediately after that, the features belonging to γ_{k-1} are triangulated, and therefore, added to λ_k for the next stage. On the other hand, they are moved from the partially active to fully active features list.

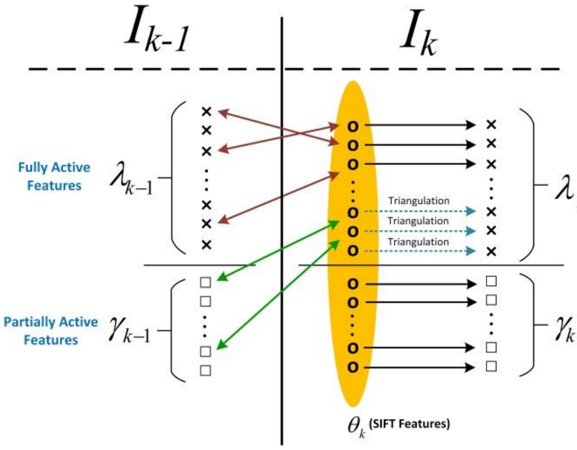

Figure 5. Overall scheme of adopted strategy followed to manage fully and partially active features.

Conversely, if the cardinality of FA_k is less than the aforementioned threshold, to recover more accurate pose parameters, we must provide more correct matchings. Doing so, the features belonging to λ_{k-1} that are not matched to any member of θ_k are moved to the new image using the homography matrix obtained from the correspondence refinement routine applied in the previous step. Some of these moved features may appear outside the image boundaries, which will be discarded. Moreover, the moved features may not accurately coincide with their true location but they can be searched within a window centred at the moved feature (blue window). Since the images are wide-baseline, searching for a precise location of matching feature within this window using simple patch-based similarity measures may lead to erroneous results.

As depicted in figure 6, u_1 is moved to $u_1^{'}$ using the homography matrix, while u_2 is its true correspondence. Hence, in order to obtain correct matchings, a square patch around the feature in I_{k-1} is warped using the homography matrix (red patch), and then this warped patch is searched in the foregoing window using normalized cross correlation.

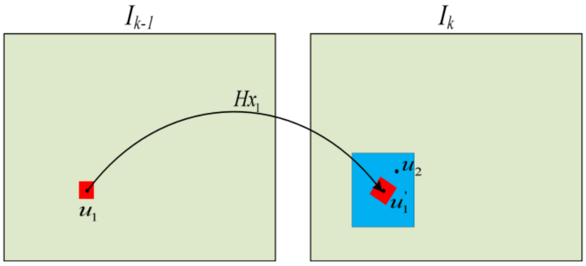

Figure 6. Obtaining correspondence based on homography matrix.

3.5. Pose estimation

After determination of matched points, we proceed to estimate the camera pose parameters. As explained in the algorithm outline, the camera orientation and translation for each incoming image is estimated directly with respect to the world referential system. As illustrated in figure 7, given a set of 3D-2D feature correspondences, we aim at finding camera pose parameters embedded in the camera projection matrix. Let X_w be the world coordinate of a scene point and u be its projection on image plane; then (5) holds.

$$\lambda u = PX_w = K(RX_w + T) \text{, with}$$

$$\mathrm{K} = \begin{bmatrix} \alpha_x & \gamma & u_0 \\ 0 & \alpha_y & v_0 \\ 0 & 0 & 1 \end{bmatrix} \tag{5}$$

where, P is the camera matrix, and R *and* T are the rotation matrix and translation vector, respectively. K is the calibration matrix that contains intrinsic parameters of the camera. α_x, α_y represent the focal length in terms of pixels, and γ is the skew coefficient between the x, y axes and is often zero. u_0, v_0 are the principal point of the camera, which would be ideally at the centre of the image.

In this paper, in order to estimate the parameters of the camera pose in each step, the EPnP method, which has been proposed by Lepetit et al. [30] is used. EPnP is a non-iterative method with computation complexity of order O(n). As most of the solutions to the PnP problem, it tries to estimate the coordinate of reference points in the camera coordinate system. Then the orientation and translation of the camera with respect to the world coordinate is computed based on a series of 3D-3D matchings using the solutions proposed for absolute orientation problem.

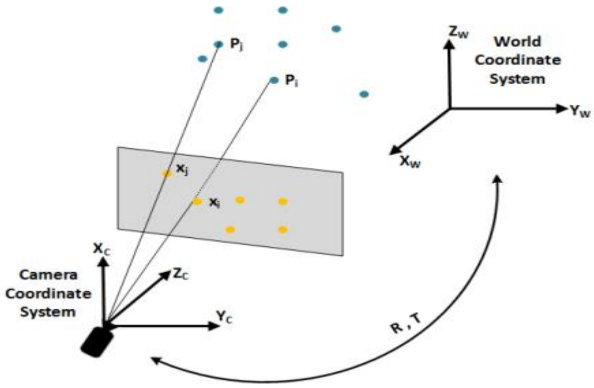

Figure 7. Camera pose estimation using 3D-2D feature points matchings.

3.6. Depth Estimation for new features

Given a feature correspondence $u_i \leftrightarrow u_{i+1}$ and camera poses encoded in camera projection matrices P_i and P_{i+1}, we are going to estimate the 3D coordinate of the associated features in the world coordinate system. According to (5), we have:

$$\lambda_i u_i = P_i X_w = \begin{pmatrix} P_{i,1}^T \\ P_{i,2}^T \\ P_{i,3}^T \end{pmatrix} X_w \,, \tag{6}$$

where, u_i, X_w are in homogenous coordinate, and $P_{i,1}^T, P_{i,2}^T, P_{i,3}^T$ are rows of the camera matrix P_i. Expanding (6), three equations with respect to unknown components of X_w are constructed, which are not linearly independent. Actually, two of them are independent, as given in (7).

$$\begin{cases} P_{i,3}^T x_i = P_{i,1}^T X_w \\ P_{i,3}^T y_i = P_{i,2}^T X_w \end{cases} \tag{7}$$

The same equations hold for u_{i+1}, as follow:

$$\begin{cases} P_{i+1,3}^{T} x_{i+1} = P_{i+1,1}^{T} X_{w} \\ P_{i+1,3}^{T} y_{i+1} = P_{i+1,2}^{T} X_{w} \end{cases} \qquad (8)$$

Putting together (7) and (8) and writing them in the matrix form, a linear system with four equations in the form of AX = 0 is obtained. This matrix equation can be solved using Singular Value Decomposition (SVD). It is worth noting that if a feature appears in more than two images, then the number of equations in the AX = 0 equation increase by the number of two for any added image. Considering the appearance of a feature in more than two images, the estimated depth for the corresponding point in the scene is more robust.

The above computations are applied to all new feature correspondences that are selected for inclusion in fully active features. This increases the possibility of finding enough matchings for the next incoming image.

4. Experimental results

We used a freely moving hand-held camera to capture images of a calibrated scene. The captured images were selected so that they properly represented a wide-baseline situation. Resolution of the captured images was 480 × 640 pixels and the algorithm works with greyscale images. It was assumed that the camera was calibrated in advance. The camera calibration was performed utilizing a flexible technique presented by zhengyou [31]. To this end, a collection of images of a chessboard with a known size taken from different viewpoints were used to estimate the intrinsic parameters of the camera. The correspondence between corners of chessboard cells and their projection on each image were then detected. Thereafter, the internal parameters of the camera were estimated by means of a closed-form solution using the correspondences obtained between the planar model and its image. The parameters obtained were then refined using a non-linear refinement based on the maximum likelihood.

A significant problem in evaluating the accuracy of the camera pose estimation methods is the lack of ground-truth data. Obtaining true pose of a moving camera w.r.t. world coordinate system is not a simple task. Using an accurate motion capture system with multiple high speed cameras is a good choice for generation of the ground-truth data. As an example of this method, Sturm et al. [32] have employed a motion capture system to construct a benchmark for the evaluation of RGB-D SLAM systems. It is also possible to generate the translation part of the camera pose manually. Davison et al. [33] have used a hand-held camera equipped with a plump-line of known length and a hanging weight skimmed to a pre-prepared rectangular track on a cluttered desk to measure the ground-truth 3D coordinate of camera at corners of track. It is clear that measuring orientation of the camera manually is not very accurate. In order to overcome this limitation, a marker-based method was employed to generate the Ground-Truth data for camera pose. Calculation of the camera pose parameters is accompanied by correspondence of easily detectable marker points on a planar surface and their projections on image plane. In our experiments, the scene was a computer desk cluttered with various objects. A planar chessboard pattern (our marker) was stuck on it that was used for calculation of ground-truth camera pose.

At the beginning, for the first two frames, the camera pose parameters were calculated using planar chessboard markers. From the third frame onwards, estimation of camera pose parameters was carried out exploiting the natural features that were correctly matched as explained earlier.

Figure 8(a) shows the visibility of the extracted features in the input images. As it could be seen, most of the features were visible only in small numbers of images (four images in our experiment). In figure 8(b), the number of matched features before refinement after refinement and the matchings with a known depth is shown. It is obvious that the number of refined matchings is less than the number initial matchings and greater than the number of matchings with a known depth, which is an expected result.

Figure 9 illustrates the trajectory of camera in a 3D space as well as its projection on the XY plane. In spite of getting no information from the environment, the camera was tracked with sufficient precision, and its pose was estimated very close to the ground-truth data. Assume that $t_{est}^{k}, t_{true}^{k}$ is the estimated and Ground-Truth translation part of camera pose and e_{t}^{k} is the associated error computed for I_k, as given in (9).

$$e_{t}^{k} = abs(t_{est}^{k} - t_{true}^{k}) = (e_{t,x}^{k}, e_{t,y}^{k}, e_{t,z}^{k})^{T} \qquad (9)$$

where, *abs(.)* denotes the absolute value function. Similarly, $r_{est}^{k}, r_{true}^{k}, e_{r}^{k}$ are defined for the rotation part of camera pose. It is worth noting that the components of rotation error were given in Euler angle representation and measured in radian.

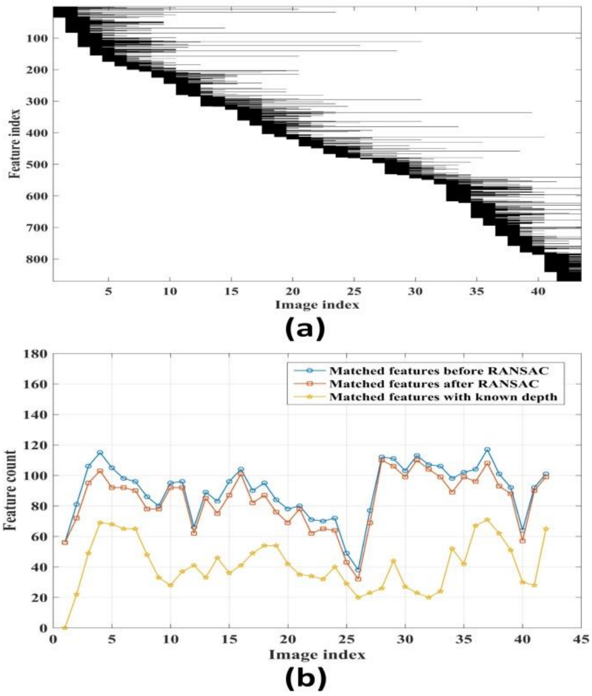

Figure 8. (a) Visibility of extracted features in images (b) Number of matched features before RANSAC after RANSAC and those matched with features whose depth is known.

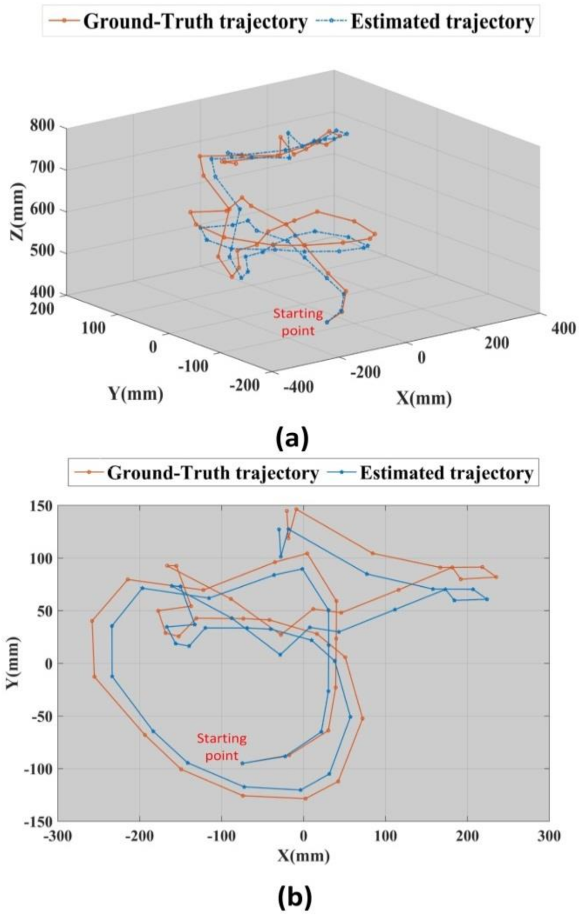

Figure 9. Ground-Truth and estimated trajectory of camera (a) in 3D and (b) projection on XY plane.

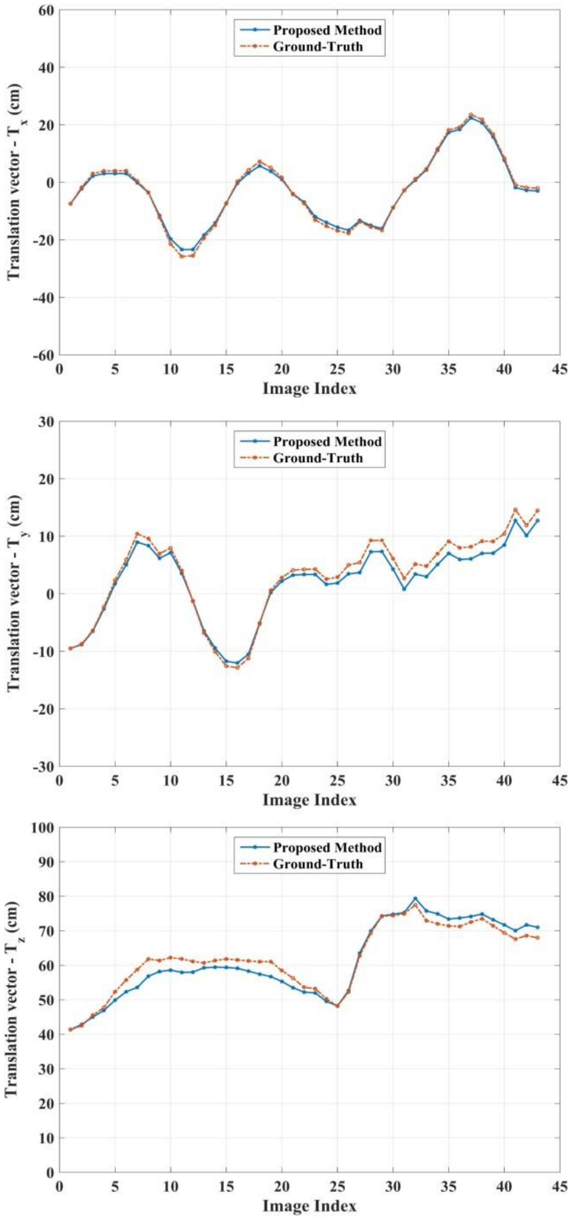

Figure 10. Estimated camera translation vector against Ground-Truth data.

Accordingly, the statistics of translation and rotation errors over all images are detailed in table 1. In figure 10, the translation components of camera pose are visualized against the computed Ground-Truth data. As it is shown, camera pose drift is negligible and the true trajectory of the camera has properly been followed.

Figure 11(a) shows the relative translation of camera center between successive images obtained from the Ground-Truth data. Figure 11(b) depicts the number of refined matchings. As illustrated in these two figures, there is a close relationship between the number of refined matchings and the translation part of camera relative pose. On the other hand, with increase in the distance of camera center in two consecutive images, the number of correct matchings was reduced.

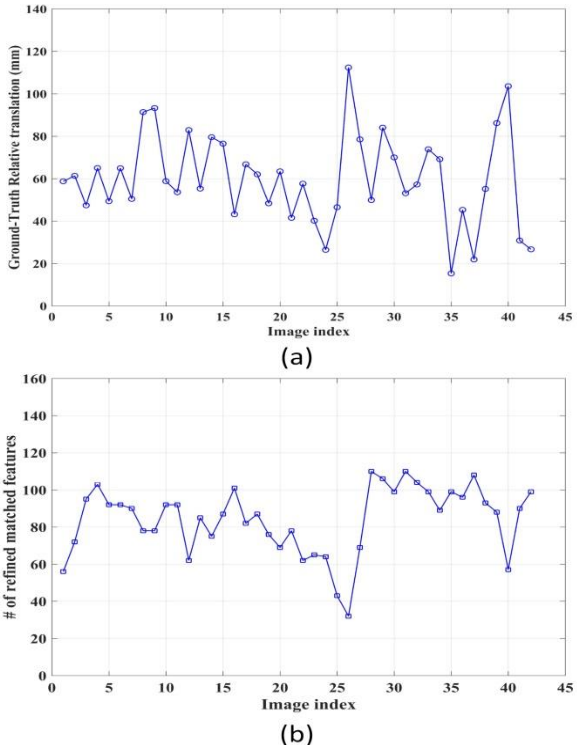

Figure 11. (a) Ground-Truth relative translation (b) Number of refined matched features between successive frames.

Table 1. Tranlation and rotation error.

	Translation error (mm)			Rotation error (radian)		
	$e_{t,x}$	$e_{t,y}$	$e_{t,z}$	$e_{r,x}$	$e_{r,y}$	$e_{r,z}$
Mean	8.41	11.89	21.46	0.17	0.26	0.08
Std	5.08	6.79	13.34	0.12	0.11	0.07
Min	0.94	0.34	0.13	0.03	0.01	0.01
Max	24.19	21.05	51.46	0.51	0.46	0.29

5. Conclusions and future works

In this work, a camera pose estimation approach was proposed for a sequence of wide-baseline images. It was considered that the camera was calibrated, and the overlapping area of the successive images was enough for acquiring a sufficient number of corresponding feature points. In the reported work, the experiments show that at least 60% of the consecutive images should be overlapped to ensure that a sufficient number of matchings are obtained.

Finding feature correspondences is the main challenge. This challenge is due to the inherent nature of wide-baseline images, in which the feature points have considerable displacement in consecutive images. In the reported work, with the exception of the first two images, no additional information about pose of the camera or position of any landmark in the scene is fed into the system. For each new image, pose of the camera was estimated according to a set of 3D-2D correspondences.

A problem that should be kept in mind is that when the number of images increases, the cumulative error for orientation and translation of camera will increase as well. If the system receives no information from the environment, then at a point in the future the error will overshoot, and as a result, the trajectory of camera undergoes an uncontrolled drift. In order to overcome this problem, it is required to either acquire some information from the scene or try to close the loop. We planned to consider the latter case in our future works. One can also investigate other feature point extractors other than SIFT and then compare the results.

References

[1] Massart, D. L., Kaufman, L., Rousseeuw, P. J. & Leroy, A. (1986). Least median of squares: a robust method for outlier and model error detection in regression and calibration. Analytica Chimica Acta, vol. 187, pp. 171-179.

[2] Fischler, M. A. & Bolles, R. C. (1981). Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Communications of the ACM, vol. 24, pp. 381-395.

[3] Ma, Y., Soatto, S., Kosecka, J. & Sastry, S. S. (2003). An Invitation to 3-D Vision: From Images to Geometric Models. Berlin, Heidelberg. New York: SpringerVerlag.

[4] Triggs, B., McLauchlan, P., Hartley, R. & Fitzgibbon, A. (1999). Bundle Adjustment — A Modern Synthesis. International workshop on vision algorithms, Corfu, Greece, 1999.

[5] Kümmerle, R., Giorgio, G., Strsdat, H., Konolige, K. & Burgard, W. (2011). g2o: A General framework for Graph Optimization. IEEE international Conference on Robotics and Automation, Shanghai, China, 2011.

[6] Kelein, G. & murray, D. (2007). Parallel Tracking and Mapping for Small AR Workspaces. 6th IEEE and ACM International Symposium on Mixed and Augmented Reality, Nara, Japan, 2007.

[7] Endres, F., Hess, J., Sturm, J., Cremers, D. & Burgard, W. (2014). 3-D Mapping with an RGB-D Camera. IEEE Transactions on Robotics, vol. 29, no. 1, pp. 177-187.

[8] Engel, J., Schöps, T. & Cremers, D. (2014). LSD-SLAM: Large-Scale Direct Monocular SLAM. 13th European Conference on Computer Vision, Zurich, Switzerland, 2014.

[9] Jain, S. & Neumann, U. (2006). Real-time Camera Pose and Focal Length Estimation. 18th International Conference on Pattern Recognition(ICPR), Hong Kong, China, 2006.

[10] Maidi, M., Ababsa, F., Mallem, M. & Preda, M. (2015). Hybrid tracking system for robust fiducials registration in augmented reality. Signal, Image and Video Processing, vol. 9, no. 1, pp. 831-849.

[11] Kim, J.-S. & Hong, K.-S. (2007). A recursive camera resectioning technique for off-line video-based augmented reality. Pattern Recognition Letters, vol. 28, no. 7, pp. 842-853.

[12] Lee, S.-H. (2014). Real-time camera tracking using a particle filter combined with unscented Kalman filters. Journal of Electronic Imaging, vol. 23, no. 1, pp. 013029-013029.

[13] Herranz, F., Muthukrishnan, K. & Langendoen, K. (2011). Camera pose estimation using particle filters. International Conference on Indoor Positioning and Indoor Navigation (IPIN), Guimaraes, Portugal, 2011.

[14] Horn, B. K. P. (1987). Closed-form solution of absolute orientation using unit quaternions. Journal of the Optical Society of America A, vol. 4, pp. 629-642.

[15] Arun, K. S., Huang, T. S. & Blostein, S. D. (1987). Least-Squares Fitting of Two 3-D Point Sets. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 9, no. 5, pp. 698-700.

[16] Horn, B. K. P., Hilden, H. M. & Negahdaripour, S. (1998). Closed-Form Solution of Absolute Orientation Using Orthonormal Matrices. Journal of Optical Socity of America, vol. 5, no. 7, pp. 1127-1135.

[17] DeMenthon, D. & Davis, L. S. (1992). Exact and approximate solutions of the perspective-three-point problem. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 14, no. 11, pp. 1100-1105.

[18] Long, Q. & Zhongdan, L. (1999). Linear N-point camera pose determination. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 21, no. 8, pp. 774-780.

[19] Ababsa, F.-e. & Mallem, M. (2004). Robust camera pose estimation using 2d fiducials tracking for real-time augmented reality systems. ACM SIGGRAPH international conference on Virtual Reality continuum and its applications in industry, Nanyong, Singapore, 2004.

[20] Maidi,M., Didier, J.-Y., Ababsa, F. & Mallem, M. (2010). A performance study for camera pose estimation using visual marker based tracking. Machine Vision and Applications, vol. 21, no. 3, pp. 365-376.

[21] Yoon, J.-H., Park, J.-S. & Kim, C. (2006). Increasing Camera Pose Estimation Accuracy Using Multiple Markers. Advances in Artificial Reality and Tele-Existence, Hangzhou, China, 2006.

[22] Xu, K., Chia, K. W. & Cheok, A. D. (2008). Real-time camera tracking for marker-less and unprepared augmented reality environments. Image and Vision Computing, vol. 26, no. 5, pp. 673-689.

[23] Dong, Z., Zhang, G., Jia, J. & Bao, H. (2014). Efficient keyframe-based real-time camera tracking. Computer Vision and Image Understanding, vol. 118, pp. 97-110.

[24] Hartley, R. & Zisserman, A. (2003). Multiple View Geometry in Computer Vision. 2nd ed. New York, NY, USA. Cambridge University Press.

[25] Harris, C. & Stephens, M. (1988). A combined corner and edge detector. Alvey vision conference, Manchester, UK, 1988.

[26] Smith, S. & Brady, J. M. (1997). SUSAN—A New Approach to Low Level Image Processing. International Journal of Computer Vision, vol. 23, no. 1, pp. 45-78.

[27] Lowe, D. (2004). Distinctive Image Features from Scale-Invariant Keypoints. International Journal of Computer Vision, vol. no. 2, pp. 91-110.

[28] Bay, H., Ess, A., Tuytelaars, T. & Gool, L. (2006). Speeded-Up Robust Features (SURF). Computer Vision and Image Underatanding, vol. 110. No. 3, pp. 346-359.

[29] Leutenegger, S., Chli, M. & Siegwart, R. Y. (2011). BRISK: Binary Robust invariant scalable keypoints. International Conference on Computer Vision, Barcelona, Spain, 2011.

[30] Lepetit, V., Moreno-Noguer, F. & Fua, P. (2009). EPnP: An Accurate O(n) Solution to the PnP Problem. International Journal of Computer Vision, vol. 81, no. 1, pp. 155-166.

[31] Zhengyou, Z. (2000). A flexible new technique for camera calibration. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 22, no. 11, pp. 1330-1334.

[32] Sturm, J., Engelhard, N., Endres, F., Burgard, W. & Cremers, D. (2012). A benchmark for the evaluation of RGB-D SLAM systems. International Conference on Intelligent Robots and Systems (IROS), Vilamoura-Algarve, Portugal, 2012.

[33] Davison, A. J., Reid, I. D., Molton, N. D. & Stasse, O. (2007). MonoSLAM: Real-Time Single Camera SLAM. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 29, no. 6, pp. 1052-1067.

نشریه هوش مصنوعی و داده کاوی

تخمین موقعیت و جهت دوربین در محیطهای ناشناس به کمک دنباله‌ای از تصاویر خط پایه عریض تک چشمی

سیدعلی حسینی و پیمان کبیری*

دانشکده مهندسی کامپیوتر ، دانشگاه علم و صنعت ایران ، تهران ، ایران.

ارسال ۲۰۱۶/۱۱/۰۲؛ بازنگری ۲۰۱۷/۰۱/۳۱؛ پذیرش ۲۰۱۷/۰۵/۱۰

چکیده:

در این پژوهش یک راهکار مبتنی بر ویژگی برای تخمین موقعیت و جهت دوربین در دنباله‌ای از تصاویر خط پایه عریض پیشنهاد شده است. تخمین موقعیت و جهت دوربین یک مساله مهم در بسیاری از کاربردهای بینایی ماشین و رباتیک نظیر واقعیت افزوده و نقشه‌سازی و مکان‌یابی همزمان دیداری می‌باشد. در روش ارائه شده تصاویر گرفته شده توسط یک دوربین در محیطی به اندازه یک اتاق و با حداکثر عمق حدود ۳ تا ۴ متر، ردیابی می‌شوند. سامانه پیشنهاد شده می‌تواند در محیطهای ناشناس و بدون دریافت اطلاعاتی از محیط مورد استفاده قرار بگیرد. البته برای مقداردهی اولیه عمق ویژگی‌های استخراج شده در دو تصویر اول، از مقدار عمق تعدادی ویژگی با عمق مشخص استفاده شده است. فرایند تخمین موقعیت و جهت دوربین صرفا با استفاده از ویژگی‌های طبیعی استخراج شده از صحنه انجام می‌شود. در تصاویر خط پایه عریض برخلاف فریم‌های متوالی یک تصویر ویدئویی مقدار جابجایی نقاط ویژگی قابل توجه بوده و لذا تعیین نقاط متناظر با کمک روش‌های مبتنی بر وصله امکان‌پذیر نیست. به همین جهت از یک روش ترکیبی برای تعیین تناظرها در تصاویر متوالی استفاده شده است. در این روش ابتدا تناظرهای احتمالی با استفاده از میزان مشابهت بردار توصیفگر ویژگی‌ها بدست آمده و سپس با استفاده از الگوریتم توافق عام نمونه تصادفی، تناظرهای نادرست حذف می‌شوند. راهکار ارائه شده بر روی تصاویر واقعی گرفته شده با کیفیت ۶۴۰ × ۴۸۰ پیکسل آزمایش شده و به طور میانگین خطای موقعیت دوربین کمتر از ۲ سانتیمتر بوده که نشان از دقت و کارایی بالای آن است.

کلمات کلیدی: تخمین موقعیت و جهت دوربین، استخراج ویژگی، تناظریابی ویژگی‌ها، تعدیل دسته‌ای، تخمین عمق.